Information Retrieval

Varun Jagadeesh

Homework - 3

Problem - (1)

Query tranformation

a) electric car → electric car cars

Query Based stemming

Stemming is a process of reducing derived words to their word stem Query based stemming is a technique where the query is expanded using various word variants. This is an expansion technique and it is used to increase the flexibility of the search engine and the effectiveness. The query is expanded using word variants.

b) facebook-linq → facebook linq

Spell check

This is one of the IR techniques which is used to correct the spelling if the user has entered wrong spelling. The suggestions will be given that is not present in the dictionary. This can be done by comparing word in entered query to words in dictionary. Spell correction is detected using a common Levenshtein edit distance function. This function maps well to a spelling correction a user would typically make, because it tracks the number of character edits between two queries. There are many techniques which can be used to speed up this process. This is one of the most important part because 10% of web queries have spelling error.

c) middle east turmoil → middle east turmoil syria iraq

Pseudo Relevance Feedback

The expansion of query takes place in such a way that it is relevant to the user. It takes the result that are initially returned from a given query and then to gather user feedback, to use whether the results are relevant or not to perform a new query. The ranking of documents takes place which are relevant. Words that frequently occur in the document may be considered in the expansion of the query.

d) new york times subscription → "new york times" subscription

Query segmentation

The query is broken into important chunks. The possible approaches which can be used are as follows.

Treat each term as a concept, Treat every adjacent pair of terms as concept

Treat all terms within a noun phrase chunk as a concept.

Treat all terms that occur in common queries as a single concept

e) tapas → tapas Cambridge Massachusetts

Local Search

The location is taken into consideration. This uses geographical information to modify the ranking of search results

location can be derived from the query text or where the query is generated. The ranking of web pages is done using location information in addition to text and link based features.

Problem - (2)

Language Modelling

a) Smoothing is a technique for estimating probabilities for missing words This is done to avoid

- estimation problem
- overcoming data sparsity.
- lower the probability estimates for words in document text & assign left over probability to estimates for words not in document text By doing smoothing in the query likelihood retrieval model the ranking of document takes place by the probability that query text could be generated by document language model. This is a topical relevance when the document & query are on same topic and the probability of query generation is the measure of how likely the document & query are same.

Disadvantage

The score given by query likelyhood model for $P(Q|D) = 0$ if there are any words present in the query are missing.

b) The likelihood of a query Q with Jelinek-Mercer smoothing

Rank score. $p(Q|D) = \prod_{i=1}^{n} [(1-\lambda) \frac{f_{q_i,D}}{|D|} + \lambda \frac{c_{q_i}}{|C|}]$

For convenience

$$\log p(Q|D) = \sum_{i=1}^{n} \log \left((1-\lambda) \frac{f_{q_i,D}}{|D|} + \lambda \frac{c_{q_i}}{|C|}\right)$$

As λ approaches zero.

Small values of λ produce less smoothing and thus the query tend to act like Boolean AND). The relative weighting of words is measured by the maximum likelyhood estimate & will be important in determining the score.

As λ approaches 1

The relative weighting will be less important & the query acts more like a Boolean OR.

λ value of 0.1 work well for short queries

λ value of 0.7 work well for long queries.

short queries tend to contain only significant words & a low value of λ will favour documents that contain all query words long queries missing a word is much less important and a high λ places more emphasis on documents that contain a number of the high-probability words.

c) The user indicates which documents are interesting i.e the relevant documents and non relevant documents. Based on this information the system automatically reformulates the query by adding terms & reweighting the original terms & a new ranking is generated using modified query. The words that occur frequently in relevant than in non-relevant.

This idea is used in the technique of pseudo relevance feedback where instead of asking the user to identify relevant documents the system assumes that the top ranked documents are relevant.

Words that occur frequently in these documents may then be used to expand the initial query.

The expansion terms generated by pseudo relevance feedback will depend on the whole query. Since they are extracted from documents ranked high for that query but the quality of expansion will be determined by how many of top ranked document in the ranking are relevant.

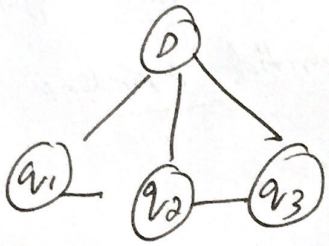
There is improvement in effectiveness. But like all pseudo relevance feedback techniques, these improvement are not consistent and some queries produce worse rankings

Problem - (3)

a) Query - "Russian president Vladimir Putin"

we know that

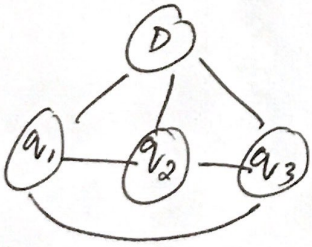
sequential dependence model



This is a bigram language model when only bigram will be considered in sequential dependence model

Bigrams - Russian president
president Vladimir
Vladimir Putin

Full dependence model



This has all bigrams that are in sequential dependence model and all trigrams, quadgrams are considered in full dependence model and these are not present in sequential dependence model.

Bigram - Russian president
president Vladimir
Vladimir Putin

Trigram - Russian president Vladimir
president Vladimir putin

Quad - Russian president Vladimir putin

b) In Naive Bayes classification model as in BM25 and unigram terms are independent of each other given the document classification

If we want to add bigram feature to Naive Bayes classifier for positive vs negative review, sequential dependence will be enabled and thus

'martin' 'charlie' 'sheen' - 'martin charlie'
'charlie sheen' will be added making the model powerful compared to any other models.

p('martin'|'sheen') & p('martin') in collection of positive reviews are equal as they are independent

Problem - (4)

a) Focus on top documents

Users tend to look at top part of ranked result list to find relevant documents

For ex: users tend to look at first page or two in web search. Navigational search or question answering have just single relevant document. Instead the focus of an effective measure should be on how well search engine does at retrieving relevant document at a very high ranks.

DCG (Discounted Cumulative gain) has become a popular measure for evaluation of web search & related applications based on these assumptions

→ Highly relevant documents are more useful than marginally relevant documents.

→ The lower the ranked position of a relevant document the less useful it is for the user as it is less likely to be examined.

These can lead to an evaluation that uses graded relevance as the measure of usefulness or gains from examining the document. Gain is accumulated at top of ranking & may be reduced at low ranks.

DCG is defined by

$$DCG_p = rel_1 + \sum_{i=1}^{p} \frac{rel_i}{\log_2 i}$$

b) MAP is the most widely used in evaluation as it is based on average precision because it assumes the user is interested in finding relevant documents for each query. It provides a very succinct summary of effectiveness of a ranking algorithm over many queries which is useful at times but disadvantage is that too much information is lost in this process.

Retrieving relevant documents at low ranks will decrease MAP
As the number of documents is increased, precision decreases. in the calculation of precision for top ranked documents. While retrieving the top ranked documents if we retrieve relevant documents at low ranks number of documents increases as non relevant documents can be retrieved with documents at low ranks. By doing this the average precision decreases and hence there there will be a decrease in MAP

c) Retrieval experiment generate data such as average precision value
In order to decide whether this data shows that there is a meaningful difference between two retrieval algorithm & significance tests are needed. Every significance test is based on a null hypothesis
In the case of typical retrieval experiment we are comparing value of an effectiveness measure for rankings produced by two algorithm.
Null hypothesis is that there is no effectiveness between two algorithm. Though type is alternate hypothesis that there is difference.
Given A & B
A → base line algorithm
B → new algorithm.
We try to show effectiveness of B better than A
Procedure for comparing two retrieval algorithm using a particular set of queries & significance test is as follows.
→ Compute the effectiveness measure for every query for both rankings
→ Compute test statistic based on comparison of effectiveness measure for each query
→ The test statistic is used to compute a P value
Small P value → null hypothesis is false.
→ Null hypothesis is rejected in favor of alternate hypothesis.